National **A**eronautics and **S**pace **A**dministration
Langley Research Center

Cloud-**A**erosol **L**idar and **I**nfrared **P**athfinder **S**atellite **O**bservations **(CALIPSO)**

SOFTWARE QUALITY ASSURANCE MANAGEMENT PLAN

Title: Software QA Management Plan	Date: January 2002	Document Owner:
Document No.: PC-PRJ-513	Version: 1.0	Leslie Johnson

Signature Page

Prepared by:

Leslie J. Johnson, LaRC
Software Quality Assurance Manager

Date: 2/27/02

Concurred by:

Brent N. Heard, LaRC
Mission Assurance Manager

Date: 2/27/02

Concurred by:

Robert F. Estes, LaRC
Mission Software Manager

Date:

Kevin Brown, LaRC
Payload Segment Manager

2/27/02

Jim Wells, LaRC
Launch Segment Manager

Mary Beth Wusk, LaRC/GATS
Mission Operations Manager

Victor Lucas, LaRC
Ground Segment Manager

Kim Cannon, LaRC
Science Segment Manager

Approved by:

John F. Rogers, LaRC
CALIPSO LaRC Project Manager

Date: 2/27/02

Bruno Belon, CNES
CALIPSO CNES Project Manager

Date: 3/21/02

REVISION HISTORY TABLE

Revision	Date	Description
Version 0.1	Jan. 5, 2001	Draft Release for comments
Version 1.0	Jan. 30, 2002	Name change – PICASSO-CENA to CALIPSO

Table of Contents

Section Page

1. INTRODUCTION

This Software Quality Assurance Management Plan describes the organization and procedures to be used by National Aeronautics and Space Administration (NASA) Langley Research Center (LaRC) to ensure that software quality assurance (SQA) is performed throughout the duration of the Cloud-Aerosol Lidar and Infrared Pathfinder Satellite Observations (CALIPSO) project and to provide confidence to management that approved processes are being followed, that high-quality products are being produced, and that the software activities comply with the application requirements and technical standards.

Software quality assurance management is essential to improving the practice of software engineering and management and thus constitutes an integral part of the software engineering process.

1.1 Identification

This document is identified as the NASA LaRC Software Quality Assurance Management Plan (SQA-MP) for the CALIPSO project. This plan establishes the policies, standards, procedures, and practices for software assurance of all computer software for applications developed for the CALIPSO project. Moreover, this plan establishes a standards compliance perspective.

The CALIPSO project is organized into four segments: Ground, Launch, Satellite, and Science. SQA applies to all software activities for the segments. The CALIPSO mission is an element of the Earth System Science Pathfinders (ESSP) program managed by Goddard Space Flight Center (GSFC). CALIPSO is a joint venture of NASA and the French Centre National d'Etudes Spatiales (CNES). Ball Aerospace & Technologies Corporation (BATC) is the prime contractor for the American payload components of the satellite segment. Sodern is a French contractor for the Imaging Infrared Radiometer (IIR) instrument. Kennedy Space Center will handle the launch segment. The satellite consists of the payload instruments and their support elements (computer, data storage, telemetry and structure) and the spacecraft bus PROTEUS (Platforme Reconfigurable pour l'Observation, les Telecommunications, et les Usages Scientifiques), built by Alcatel for CNES. The ground segment includes LaRC payload operations, CNES satellite operations, and BATC-provided science data ground system. The science segment includes LaRC data management, the French data site provided by the Institute Pierre Simon Laplace and Hampton University implementation of algorithms.

This SQA-MP is a living document and as a result additions, deletions, and modifications will occur as it is utilized. It will be updated as the work proceeds and the necessity arises.

1.2 Purpose

The purpose of the SQA-MP is to ensure the quality of newly developed and modified software. This is accomplished by the following software quality assurance tasks:

Audits, Reporting and Control, and Surveys which include Process Evaluations, Spot checks / Inspections, and Code Analysis / Walkthroughs. Software quality assurance activities are also included in the Life Cycle Reviews.

1.3 Scope

The scope of this document is to define the activities to be carried out in support of software quality assurance for the CALIPSO project. This document defines the actions that provide assurance that the software-related items delivered conform to their established and contracted technical requirements. SQA will also ensure that standards, practices, procedures, and methods of the software development process are defined and applied.

Establishing standards and procedures for software development is critical, since these provide the framework from which the software evolves. Standards are the established criteria to which the software products are compared. Procedures are the established criteria to which the development and control processes are compared. Standards and procedures establish the prescribed methods for developing software; the SQA role is to ensure their existence and adequacy. Proper documentation of standards and procedures is necessary since the SQA activities of process monitoring, product evaluation, and auditing rely upon unequivocal definitions to measure project compliance.

Types of standards include: Documentation, Design, and Code. Documentation Standards specify form and content for planning, control, and product documentation and provide consistency throughout a project. Design Standards specify the form and content of the design product. They provide rules and methods for translating the software requirements into the software design and for representing it in the design documentation. Code Standards specify the language in which the code is to be written and define any restrictions on use of language features. They define legal language structures, style conventions, rules for data structures and interfaces, and internal code documentation.

Procedures are explicit steps to be followed in carrying out a process. All processes should have documented procedures.

1.4 Applicable Documents

The following list contains related documents which provide supporting information pertinent to the information in this SQA-MP:

1. Project Plan
2. Requirements Documents (System and Software)
3. Mission Assurance Management Plan
4. Project Configuration Management Plan
5. Software Quality Assurance Plans for each software development

Other plans related to the quality of software development may include the Software Management / Development Plan (SM/DP) which describes the standards, schedule and procedures to be followed by the development team, the Software Configuration Management Plan (SCMP) which describes how products of this project shall be identified, baselined, and controlled, and the Software Test Plan which describes how products of this project shall be tested.

This plan conforms to the following technical standards:

- IEEE/EIA 12207.0 Standard for Information Technology – Software life cycle process
- IEEE/EIA 12207.1 Standard for Information Technology – Software life cycle process – Life cycle data
- IEEE Std 730-1998 IEEE Standard for Software Quality Assurance Plans
- ISO 9000-3:1991 Quality management and quality assurance standards -- Part 3; Guidelines for the application of International Organization for Standardization (ISO) 9001 to the development, supply and maintenance of software
- SEI-TR-25 Key Practices of the Capability Maturity Model (CMM), Version 1.1, Carnegie Mellon University (CMU), Software Engineering Institute (SEI).
- NPD 2820.1 NASA Software Policies
- NASA-STD-2100-91 Software Documentation Standards
- NASA-STD-2201-93 Software Assurance Standard
- NASA-STD-8719.13A Software Safety
- NASA-GB-A201 Software Assurance Guidebook
- NASA-GB-001-96 Software Management Guidebook
- LMS-CP-5528 Software Planning, Development, Acquisition, Maintenance, and Operations
- LMS-CP-4754 Quality Assurance for Software Development and Acquisition

LaRC has an International Standards Organization (ISO-9000) compliant quality system known as the Langley Management System (LMS). Center Procedures (CPs) are applicable to the CALIPSO project.

2. ORGANIZATION AND RESOURCES

2.1 Organization

The LaRC Office of Mission Assurance (OMA) organization is a separate organization (i.e., separate from the engineering organization) which reports directly to the CALIPSO Project Manager. The organization is obligated to fully advise the CALIPSO Project Manager of the status of the software and the accompanying documentation.

The OMA is responsible for the fulfillment of, and for ensuring compliance with, the software quality assurance activities defined in this document. The OMA has clearly defined authority and responsibility. The OMA is responsible for ensuring that processes are applied as intended and that products are of high quality. The OMA provides senior management with an independent evaluation of the effort, and it provides a direct reporting line to the senior management of the CALIPSO project to resolve problems. The CALIPSO Project Manager can waive any request set forth in this plan provided this waiver is documented.

2.2 Resources

The OMA has the responsibility for assuring that all software development phases of the software life cycle are conducted in accordance with requirements and standards. The OMA shall select a SQA representative experienced with the activities of the software life cycle processes and that has background in software engineering principles with experience in developing software.

The OMA SQA representative will exercise oversight of all SQA planning and activity in concert with the partners' SQA engineers (SQAE). Each software development team shall have an SQAE. Sections 3 and 4 of this plan are general SQA activities that can be tailored for each software development. The SQAE should notify the OMA SQA representative of upcoming meetings/teleconferences, audits, evaluations, reviews, surveys, newly released documents and revisions, testing, acceptance, and delivery.

3. GENERAL SOFTWARE QUALITY ASSURANCE METHODS

The SQA activities shall be performed by the SQAE assigned to the software areas of the CALIPSO project.

3.1 Audits

Throughout the life cycle of the project, the SQAE performs audits to determine whether software procedures, standards, and practices have been identified and are being properly implemented. Audits also identify those areas in which additional controls and standards are required to ensure the quality of the software product. An audit may be a stand-alone activity or incorporated into a review.

3.2 Evaluating Processes and Procedures

The following software processes are evaluated by the SQAE:

- Requirement Management
  - System requirements allocated to software are controlled to establish a baseline for software engineering and management use.
  - Software plans, products, and activities are kept consistent with the system requirements allocated to software.

- Software Project Planning
  - Software estimates are documented for use in planning and tracking the software project.
  - Software project activities and commitments are planned and documented.
  - Affected groups and individuals agree to their commitments related to the software project.

- Software Project Tracking and Oversight
  - Actual results and performances are tracked against the software plans.
  - Corrective actions are taken and managed to closure when actual results and performance deviate significantly from the software plans.
  - Changes to software commitments are agreed to by the affected groups and individuals.

- Software Product Engineering
  - The software engineering tasks are defined, integrated, and consistently performed to produce the software.
  - Software work products are kept consistent with each other.

- Software Quality Assurance
  - Software quality assurance activities are planned.
  - Adherence of software products and activities to the applicable standards, procedures, and requirements is verified objectively.
  - Affected groups and individuals are informed of software quality assurance activities and results.
  - Noncompliance issues that cannot be resolved within the software project are addressed by senior management.

- Software Configuration Management
  - Software configuration management activities are planned.
  - Selected software work products are identified, controlled, and available.
  - Changes to identified software work products are controlled.
  - Affected groups and individuals are informed of the status and content of software baselines.

- Software Contract Management
  - The prime contractor selects qualified software subcontractors.
  - The prime contractor and the software subcontractor agree to their commitments to each other.
  - The prime contractor and the software subcontractor maintain ongoing communications.
  - The prime contractor tracks the software subcontractor's actual results and performance against its commitments.

3.3 Reporting and Control

3.3.1 Problem Reporting

When an anomalous condition is found in the development project life cycle or maintenance process, the individual(s) who discover the condition will document the problem. The problem will be tracked from its inception to resolution.

The SQAE has the responsibility of establishing and maintaining a problem file. Each file shall contain the problem, inclusive of checklists, notes and procedures, the corrective action replies, and any associated correspondence.

During reviews, audits, and testing, a problem shall be written against questionable items. These problems shall be answered by the appropriate engineering organization within a predefined period. The SQAE has the responsibility to monitor this process to ensure that all problems are answered correctly and completely.

3.3.2 Corrective Action

The SQAE evaluates the corrective action system for handling problems. The objective of the corrective action system is to provide a systematic method of resolving identified problems. Identification of problems may be based on:

a. contractual, company, departmental, or program procedural nonconformance;
b. unsatisfactory quality trends discovered in internal reviews and walkthroughs;
c. functional deficiencies and performance problems traceable to software errors, operator errors, testing, or inspection discrepancies;
d. unsatisfactory supplier or subcontractor performance; and
e. inconsistencies, errors, and omissions in specifications and documents.

Action items requiring corrective action may result from reviews and walkthroughs. Regardless of the source of the problem, the SQAE conducts a follow-up inspection of the corrected documents or code to verify correction of the original deficiency.

The following characteristics are essential to the corrective action system:

a. correct, complete, and accurate problem reporting;
b. careful analysis of the problems reported;
c. proper classification of the problems as to category and severity;
d. assignment of responsibility for correcting the problem;
e. determination of effective actions to be taken to remedy the problem;
f. regular, thorough analysis of trends;
g. effective, implementable recommendations resulting from trend analysis;
h. proper authorization of steps to correct problems;
i. complete records of documented actions taken;
j. re-evaluation of corrective actions after being taken;
k. tracking corrective action progress and closing out completed actions; and
l. customer visibility into the corrective action process.

3.3.3 Quality Assurance Records

The SQAE maintains records as objective evidence to substantiate that the system conforms to the requirements of the technical standards and that the customers' quality requirements are being met. In order to provide objective evidence that quality is being achieved, it is necessary that all records produced in the attainment of quality be properly maintained and catalogued for future reference. These records shall be available to the OMA SQA representative at any time.

The SQAE files may include the following records:

- Annotated copies of documents that have been the subject of SQAE Inspections, Audits and Reviews
- Minutes of Reviews
- Data and Analysis Graphics for Software Metrics
- Completed Checklists
- Internal Review Reports
- Approval Records
- Monthly SQA Reports
- Document Review Reports
- Test Observation Records
- Change Control Documentation
- Problem Reports and Corrective Action Records
- Issue Tracking Forms
- Enhancements Requests and Records on Actions taken in Response

To assist in making and evaluating the software development products and processes, basic software metrics can be used. These basic metrics are:

1) Product size, of the total software product and of each major component of that product.
2) Person-months for the total project and by project phase and for major software components.
3) Schedule time for the total project and by project phase.
4) Number of software errors made and discovered in each project development phase.
5) Number of software errors discovered after completion of the software development.
6) Software requirements volatility (i.e., the percent of software that has been modified or added after the end of the software requirements definition phase.
7) Number of tests successfully executed as a percentage of the total number of tests planned.
8) The total number software problem reports written compared with the number of software problem reports that have been closed.

The SQAE supports the collection of data for these metrics and prepares reports documenting the conclusions of the analysis of the metrics data. Graphic plots of the above metric values versus time can be used by the SQAE to reveal trends. The SQAE may also contrast the metric values obtained during the CALIPSO project with values obtained from past projects.

3.4 Surveys

3.4.1 Spot Checks / Inspections

Spot checks and inspections are short-notice reviews of the software process and product samples to monitor standards compliance. The SQAE performs spot checks with each software developer in the first few weeks of each new product development activity and provides the software developers with specific feedback on standards compliance. The SQAE role is to observe, participate as needed, and verify that the spot checks and inspections were properly conducted and documented.

3.4.2 Code Analysis / Walkthroughs

It will be necessary to conduct internal code analysis/walkthroughs prior to the reviews. The SQAE is present to lend support in interpretation of standards and contractual aspects as they apply to the subject of the code analysis / walkthrough.

A walkthrough is the procedure used by the leader of a review meeting. The walkthrough is his means of leading the audience through the product with the goal of uncovering problems, inconsistencies, etc., which would be otherwise overlooked. The walkthrough procedure will be used whenever software development products or processes are scheduled to undergo internal review. Thus, the topic of a walkthrough can be the top-level design of the software, the detailed design, the software user manual, etc.

Code Analysis verifies that the coded program correctly implements the verified design. The techniques used in performance of code analysis mirror those used in design analysis.

The SQAE role is to observe, participate as needed, and verify that the analyses and walkthroughs were properly conducted and documented.

4. PROJECT LIFE CYCLE PHASE SQA ACTIVITIES

Life cycle reviews are formalized activities which maintain visibility into the process of software development. These reviews are scheduled at the beginning of the project and represent the necessary milestones in the development process. The responsible organizations initiating the review shall ensure proper notification of impending reviews. This section contains detailed SQAE activities and software safety tasks for each life cycle review phase.

The purpose of the software safety activities is to ensure that software does not cause or contribute to a system reaching a hazardous state; that it does not fail to detect or take corrective action if the system reaches a hazardous state; and that it does not fail to mitigate damage if an accident occurs. Many of the safety analyses are iterative, taking place during the software development life cycle (i.e., the results from one phase feed the analyses of the next). As the detail of the project evolves, so does the maturity of the safety analysis.

The status and results of SQAE activities within a given life cycle phase shall be documented in the SQAE files.

4.1 Software Concept and Initiation Phase

The SQAE reviews the SM/DP to verify that it complies with all requirements and describes a practical and effective software development process. The SM/DP shall describe the project specific software products and processes, and specifically address the mechanism by which safety-critical requirements are generated, implemented, tracked, and verified. In particular, the SQAE confirms that adequate provisions have been made for error prevention and early error discovery procedures, techniques and tools.

4.2 Software Requirements Phase

The SQAE reviews the software requirements documents to make certain they are complete and consistent, and are a suitable foundation for the subsequent design and test activities. In analyzing the quality of requirements, special attention should be paid to the "testability" of each requirement. That is, if there is no straightforward way to test or inspect whether or not a requirement is met, then the statement of the requirement is deficient. This document shall also be checked for compliance with its applicable document preparation standard to verify it has the requisite format and content.

Two software safety tasks shall be performed in this phase: development of software safety requirements; and analysis of the software requirements for potential hazards. The successful development of safety requirements for the software requirements specification is essential to developing safe software and allows for safety to be built into the software early in the life cycle while it is relatively inexpensive.

The SQAE verifies that the necessary documentation and preparation steps leading up to the Software Requirements Review (SRR) have been followed.

4.3 Software Design Phase

The software design process develops the high- and low-level design that will implement the software requirements. The design process shall also include identification of safety design features and methods (e.g., inhibits, traps, interlocks, and assertions) that will be used throughout the software to implement the software safety requirements.

The SQAE verifies that the necessary documentation and preparation steps leading up to the Preliminary Design Review (PDR) have been followed. The first portion of the design phase will end with the Preliminary Design Review.

The SQAE also verifies that the necessary documentation and preparation steps leading up to the Critical Design Review (CDR) have been followed. The final portion of the design phase will end with the Critical Design Review.

The SQAE reviews the software design documents according to the SM/DP. These documents shall be examined to make certain they are in their required form and have the required content.

4.3.1 Project Documentation Review

Documentation throughout the life cycle of the project must be inspected and checked to ensure that it is correct, per the SM/DP. The following criteria shall be used:

- Compliance to documentation requirements and standards
- All necessary information has been included
- Text is clear and unambiguous
- Adequate traceability and cross-referencing has been provided and is maintained

Each phase of the life cycle generates different documents which need to be inspected. These documents shall be subjected to a SQAE inspection prior to the formal review of which they will be a part.

The following is a list of documentation:

- Project Management Plan
- Functional Requirements Document
- Software Management Plan / Safety Management Plan
- Software Development Plan
- Software Safety Requirements
- Software Preliminary Design Specification
- Software Detailed Design Specification
- User and Operations Guide
- Training Plan and Procedures
- Safety Test Plan and Procedures
- Software Test Plan and Procedures
- Site Installation Plan

4.4 Software Implementation Phase

The software implementation translates the detailed design into code in the selected programming language. The code shall implement the safety design features and methods developed during the design process. The SQAE reviews the software code and verifies that safety-critical code is commented in such a way that future changes

can be made with a reduced likelihood of invoking a hazardous state. The SQAE confirms that the implemented code is traceable to the functions specified in the design documents produced in earlier project phases and also confirms that the source code follows coding standards defined in the SM/DP. Prior to the software development organization submitting the code for software integration, the software code is to be subjected to SQAE for review.

4.5 Software Integration and Test Phase

The SQAE reviews the software and safety test plans to ensure that they describe a software test activity that thoroughly exercises the software implementation and provides evidence that all software/safety requirements have been satisfied.

The SQAE reviews the software and safety test procedures to ensure that they are consistent with the software and safety test plans and describe effective and comprehensive tests. A major criterion in evaluating these test procedures is to make certain that the descriptions of the test are sufficient for a person other than the test procedure author to conduct the test.

The SQAE witnesses selected software and safety tests and reviews the results of many additional tests that it does not witness. The purpose of this test witnessing and the test results review is to confirm that the tests were run as described in the test procedures and that the results are acceptable.

4.5.1 Testing

Testing is an activity performed by the software development organization and reviewed by the SQAE. Its primary concern is the controlled exercise of the program code using sample input cases with the objective of exposing errors. This process begins with the smallest unit of the system, continues up to, and includes installation.

In order to accomplish this task, it is necessary to generate a software test plan and software test procedures. These products must be reviewed by the responsible engineering organization manager and SQAE to ensure their completeness according to standards.

4.5.1.1 Software Test Plan

The test plan shall be subjected to a SQAE review prior to the PDR and the CDR. The test plan contains test objectives, methodologies (from unit test to installation and checkout), description of the test environment, test description, delineation of the requirements verified, an evaluation plan, and a test schedule.

The test plan is begun in the SRR and revised as the system matures. This maturation process continues until CDR. Prior to PDR and again prior to CDR this plan is submitted for the SQAE to review.

4.5.1.2 Software Test Specifications and Procedures

Software test specifications and procedures are step-by-step descriptions of the actions necessary to execute a particular test. Each test procedure shall contain the scope of the test, the applicable documents, organizational responsibilities, test environment, test data, criteria for acceptance, and the steps necessary to complete the test. Each individual step should be comprised of the user input and the expected result. All test procedures shall be subjected to a SQAE review prior to the test. The SQAE reviews the procedures to determine their completeness based on requirements and contractual agreement.

4.5.1.3 Software Test Performance

The SQAE shall be notified of all software related tests. It is the responsibility of the SQAE to note any deviation from the accepted procedures. As the test is being conducted, the software development organization shall initial each step as it is completed. Hard copies of data (logs or copies of screen images) and output media (diskette or tape) shall be baselined by the software development organization. The SQAE reviews these test results and related documents.

4.5.1.4 Software Test Assurance Reports

Upon completion of the test, a post test briefing shall be conducted by the test team. During this meeting with the test personnel, programmers, customer, and the SQAE, the results of the test are reviewed. Any problems noted during the test and any problems found in analysis shall be documented. The results of the post test briefing shall be written up by the test team and reviewed by the SQAE.

4.5.1.5 Installation and Checkout

Software, which passes formal testing, is ready for installation. The SQAE ensures that all schedules for new or modified hardware have been met and that new or modified hardware has been tested and is ready to accept the tested software. After the system has been brought up, the tested software is loaded and a rerun of the final integrated test is conducted. The results of this rerun shall be compared to the original test to determine if the system is ready for operation.

4.5.2 Verification and Validation

Software verification and validation is the process of ensuring that software being developed or changed will satisfy functional and other requirements (validation) and each step in the process of developing the software is accomplished (verification). The SQAE validates test procedures to specified requirements, verifies the test plan implementation, witnesses performance tests, and has the capability to modify the test plan to include any special customer requested testing.

4.6 Software Acceptance and Delivery Phase

Prior to the final release of a deliverable (as defined in the SCMP), it must meet the approval of the Software Manager and the leader of the SQAE. The necessary

approval is signified by including a block of signatures on the cover sheet of the Release document, or on the software delivery cover letter.

The procedure leading up to the sign-off on the deliverable by the leader of the SQAE, includes having passed the designated SQA milestones specified in the life-cycle steps. These steps are documented below:

(1) Verify that all documents or code were successfully completed, and that required signatures have been obtained and Verification Description Document is accurate.
(2) Assess the status of outstanding problems to determine if they are all closed and that corrective actions have been implemented and will be completed before acceptance. Deliverables may be released with outstanding problems with approval of the CALIPSO Project Manager.
(3) Perform an independent assessment of the deliverable (read the document or exercise the software) to form an opinion as to the quality level which will result from the release of the deliverable.
(4) If a hold on the deliverable is warranted, consult with the CALIPSO Project Manager to discuss the criticality of the timing of the release and the quality of the product to be released. The CALIPSO Project Manager shall decide whether or not to sign off on the delivery.
(5) If an impasse is reached in the decision whether or not to authorize the delivery, document this fact as an issue, and notify the next higher level of management.
(6) If the decision by the next higher level of management is to make the delivery, the delivery sign off on the “SQA Manager” line of the cover sheet shall be co-signed by the manager making the decision.

4.7 Software Sustaining Engineering and Operations Phase

4.7.1 Delivery and Installation

The SQAE confirms that all copies are identical to the copy approved as a part of the Acceptance Phase. The SQAE verifies that the installation instructions that accompany the delivery are correct and complete.

4.7.2 Maintenance

Maintenance on this project is defined to be all changes made after the first delivery. The SQAE continues to monitor the maintenance of the software products after delivery to make certain that the quality and reliability do not degrade as a result of maintenance actions. In monitoring the maintenance actions, the SQAE follows the procedures described in Section 3 and 4 of this plan. The SQAE records continue to be maintained and updated during the SQAE support of the maintenance activity. During the maintenance activity particular attention shall be paid to the continuing operation of the problem reporting and corrective action system and making certain the software configuration management procedures are still followed.

5. PROVIDER / SUBCONTRACTOR CONTROL

This section describes the SQAE procedures to evaluate provider/subcontractor products and commercially available, reusable, and government furnished software.

5.1 Sub-supplier Products

A provider/subcontractor is any outside organization developing computer program products and services for the prime contractor. The SQAE responsibilities for this area include:

- Identification of contract quality requirements to ensure inclusion within provider/subcontractor's Request For Proposals and Request For Quotations
- provider/subcontractor's surveys, inspections, and audits
- Evaluation of provider/subcontractor's system of quality assurance and control
- Review provider/subcontractor's deliverable requirements list documentation
- Audits of formal design reviews, acceptance testing, and configuration audits conducted by the provider/subcontractor
- Review contract changes/amendments to assure inclusion of or consistency with quality requirements.

5.2 Commercially Available, Reusable, and Government Furnished Software

The SQAE describes the plans for evaluation of the reliability and quality of commercially available, reusable, and Government Furnished Software.

6. ACRONYMS

This section contains an alphabetical list and definitions of all acronyms and abbreviations used in the document and any word used in a non-standard way.

BATC	Ball Aerospace Technologies Corporation
CALIPSO	Cloud-Aerosol Lidar and Infrared Pathfinder Satellite Observations
CDR	Critical Design Review
CMM	Capability Maturity Model
CMU	Carnegie Mellon University
CNES	Centre National d'Etudes Spatiales
CP	Center Procedure
EEE	Electrical, Electronic and Electromechanical
ESSP	Earth System Science Pathfinder
GSFC	Goddard Space Flight Center
ISO	International Standards Organization
LaRC	Langley Research Center
LMS	Langley Management System
NASA	National Aeronautics and Space Administration
OMA	Office of Mission Assurance
PDR	Preliminary Design Review

PROTEUS	Platforme Reconfigurable pour l'Observation, les Telecommunications et lest Usage Scientifiques
SCMP	Software Configuration Management Plan
SEI	Software Engineering Institute
SM/DP	Software Management / Development Plan
SQA	Software Quality Assurance
SQA-MP	Software Quality Assurance Management Plan
SQAE	Software Quality Assurance Engineer
SRR	Software Requirements Review
STD	Standard
SVVP	Software Verification and Validation Plan

7. CHECKLIST

The following items will be assessed by the OMA SQA representative when evaluating the SQAE activities for the CALIPSO project.

MANAGEMENT:

1. Is the SQA performed by an individual who is organizationally independent of the software project and has the skills, responsibility, authority, and organizational freedom to permit objective software product and process evaluations?
2. Is the SQA performing the following tasks?
   - Ensures that each activity identified in the software plan is performed in accordance with the plan
   - Ensures that each software product identified in the software plan is prepared and undergoes software product IV&V and corrective action as defined in the plan
   - Prepares records of software quality assurance activities and maintains those records for the life of the project
   - Provides recommendations for process and product improvement to the software manager and software team

DOCUMENTATION:

3. Is the SQA ensuring that the implementation of the software satisfies requirements, the documentation listed below is suggested:
   - Software Requirements Specification
   - Software Design Description
   - Software Verification and Validation Plan
   - Software Verification and Validation Report
   - User Documentation
   - Software Configuration Management Plan
   - Software Development Plan
   - Standards and Procedures Manual
   - Software Project Management Plan
   - Software Maintenance Manual

STANDARDS, PRACTICES, CONVENTIONS, AND METRICS:

4. Are the standards, practices, conventions and metrics to be applied identified and is it stated how compliance with these items is to be monitored and assured? The subjects covered include the basic technical, design, and programming activities involved, such as documentation, variable and module naming, programming, inspection, and testing. The following information is suggested:
   - Documentation standards
   - Logic structure standards
   - Coding standards
   - Commentary standards
   - Testing standards and practices
   - Selected SQA product and process metrics such as branch metric, decision point metric, domain metric, error message metric, and requirements demonstration metric.

REVIEWS AND AUDITS:

5. Are the technical and managerial reviews and audits to be conducted defined?
6. Is it stated how the reviews and audits are to be accomplished?
7. Is it stated what further actions are required and how they are to be implemented and verified?
8. Suggested Reviews:
   - Software Requirements Review
   - Preliminary Design Review
   - Critical Design Review
   - Software Verification and Validation Plan (SVVP) Review
   - Functional Audit
   - Physical Audit
   - In-process audits
   - Managerial reviews
   - Software Configuration Management Plan Review
   - Post-mortem Review
   - User Documentation Review

TEST:

9. Are all the tests not included in the SVVP for the software identified by the SQAP and are the methods to be used stated?

PROBLEM REPORTING AND CORRECTIVE ACTION:

10. Are the practices and procedures to be followed for reporting, tracking and resolving problems identified in the software development and maintenance process?
11. Are the specific organizational responsibilities concerned with their implementation stated?

TOOLS, TECHINIQUES, AND METHODOLOGIES:

12. Are the special software tools, techniques, and methodologies that support SQA identified, their purposes stated, and their use described.

CODE CONTROL:

13. Are the methods and facilities used to maintain, store, secure, and document controlled versions of the identified software during all phases of the software life cycle defined? This may be implemented in conjunction with a computer program library. This may also be provided as part of the SCMP.

MEDIA CONTROL:

14. Are the methods and facilities to be used to identify the media for each computer product and the documentation required to store the media stated, including the copy and restore process; and protect computer program physical media from unauthorized access or inadvertent damage or degradation during all phases of the software life cycle?

SUPPLIER CONTROL:

15. Are the provisions stated for assuring that software provided by suppliers meets the established requirements?

RECORDS COLLECTION, MAINTENANCE, AND RETENTION:

16. Identify the SQA documentation to be retained identified?
17. Are the methods and facilities to be used to assemble, safeguard, and maintain this documentation stated?
18. Has the retention period been designated?

TRAINING:

19. Are the training activities necessary to meet the needs of the SQAP identified?

RISK MANAGEMENT:

20. Are there methods and procedures specified and employed to identify, assess, monitor, and control areas of risk arising during the software life cycle?